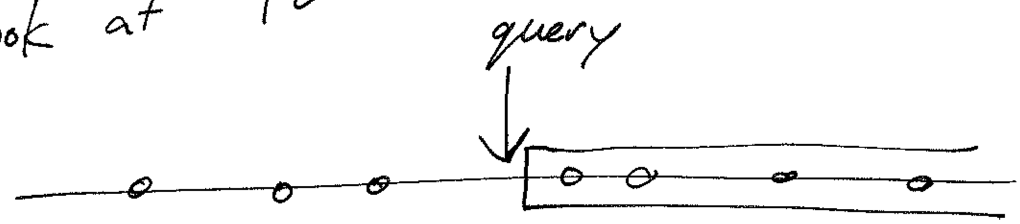
Half-space Range Counting

Input: $P \subset \mathbb{R}^2$, $|P| = n$

Queries: Given a halfspace H, return the number of points of P in H.

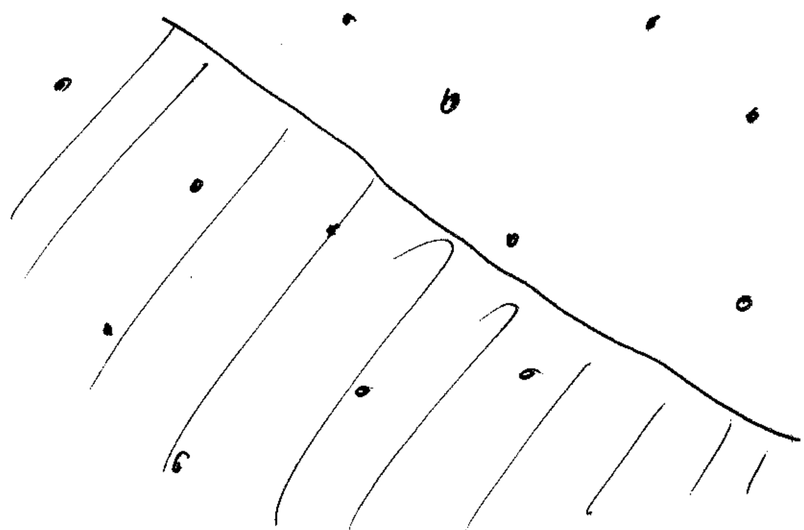
Goal: $O(\log n)$ time $O(n^2)$ space + preprocessing.

Look at 1D



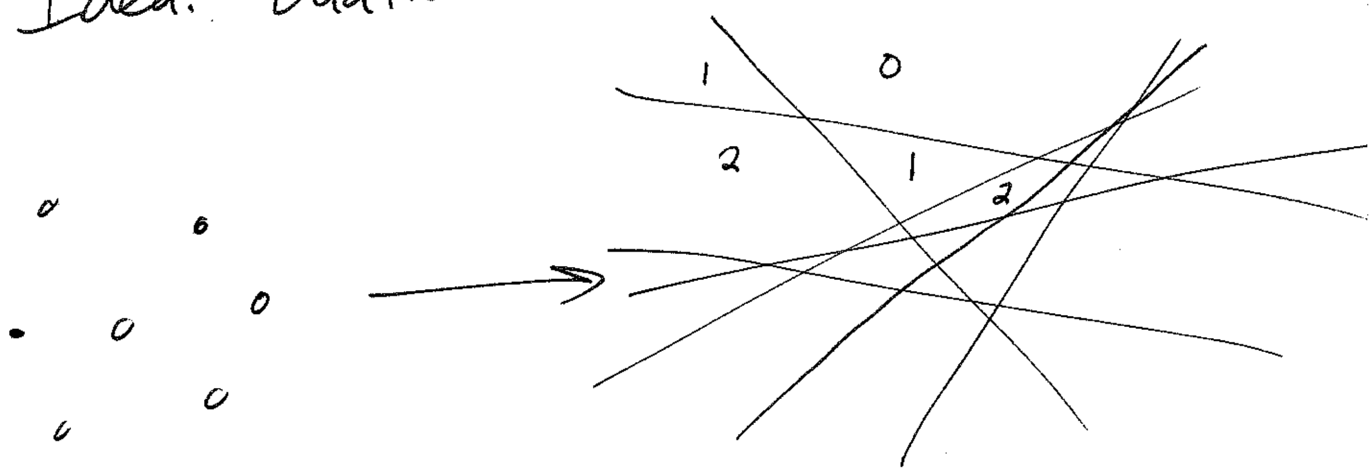
$\log(n)$ time is easy with BSTs.

What about $\mathbb{R}^2$



It's like sorting in every direction at once.

Key Idea: Dualize



Def The <u>Arrangement</u> of a set of lines L is the polyhedral complex $A(L)$ decomposing $\mathbb{R}^2$ s.t. $\displaystyle\bigcup_{\text{edges } e \in A(L)} e = \bigcup_{\ell \in L} \ell$.

Note: Every polygon in the arrangement is convex.

Assuming the queries are not bounded vertical lines, we can search for the query point (dual of a line) using the search algorithm from last time.

Store # lines above and below each cell.

Claim: Queries take $O(\log n)$ time

pf. The arrangement has $\binom{n}{2}$ vertices.
Point location in a polyhedral complex with $\binom{n}{2}$ vertices requires $O(\log \binom{n}{2}) = O(\log n)$ time using the history DAG.

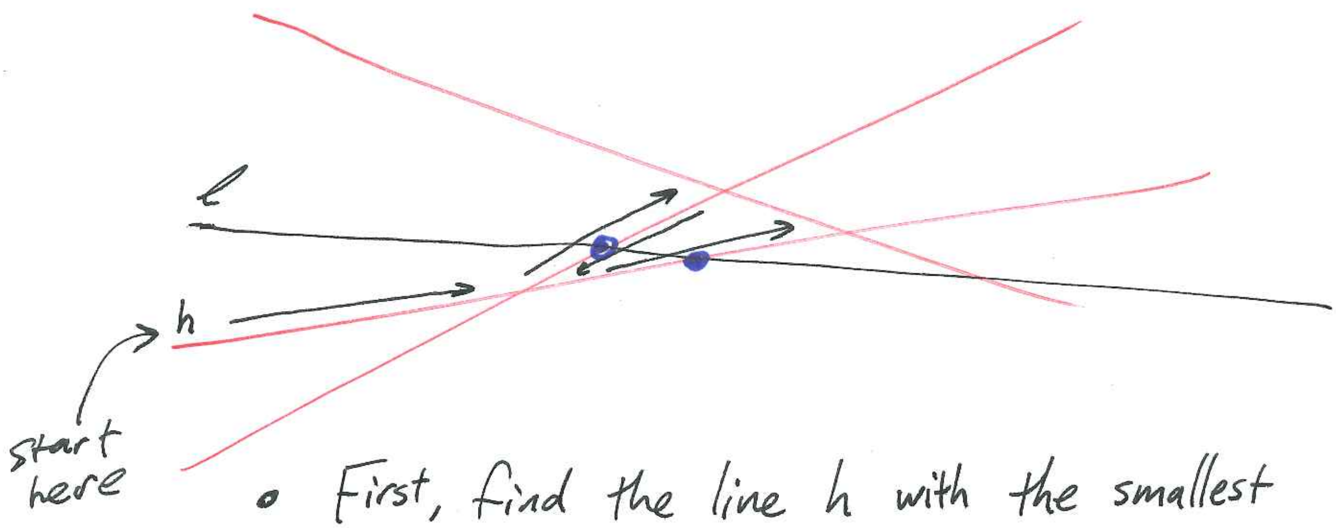
But how do we build $A(P^*)$?

Idea: Incremental Construction
- add one line at a time
- maintain a half-edge data structure.

Goal: Insert the i^{th} line in $O(i)$ time.
This will give a running time of

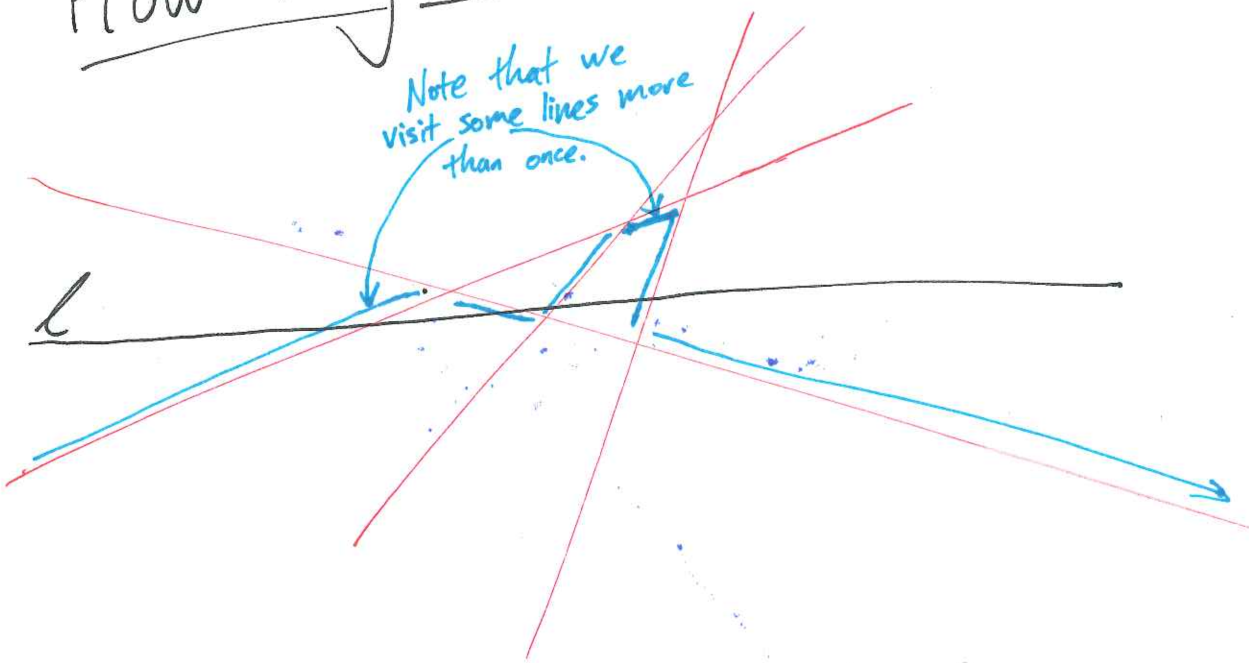
$$\sum_{i=1}^{n} O(i) = O(n^2) \text{ as desired.}$$

How to add a line



- First, find the line h with the smallest slope among the lines with slope larger than ℓ.
- Next, walk around the polygon above the leftmost segment of h.
  Walk until you cross ℓ.
  You found the first intersection.
- Cross to the face adjacent to the edge where we just intersected ℓ.
- Walk around this face until we intersect ℓ again.
- Repeat

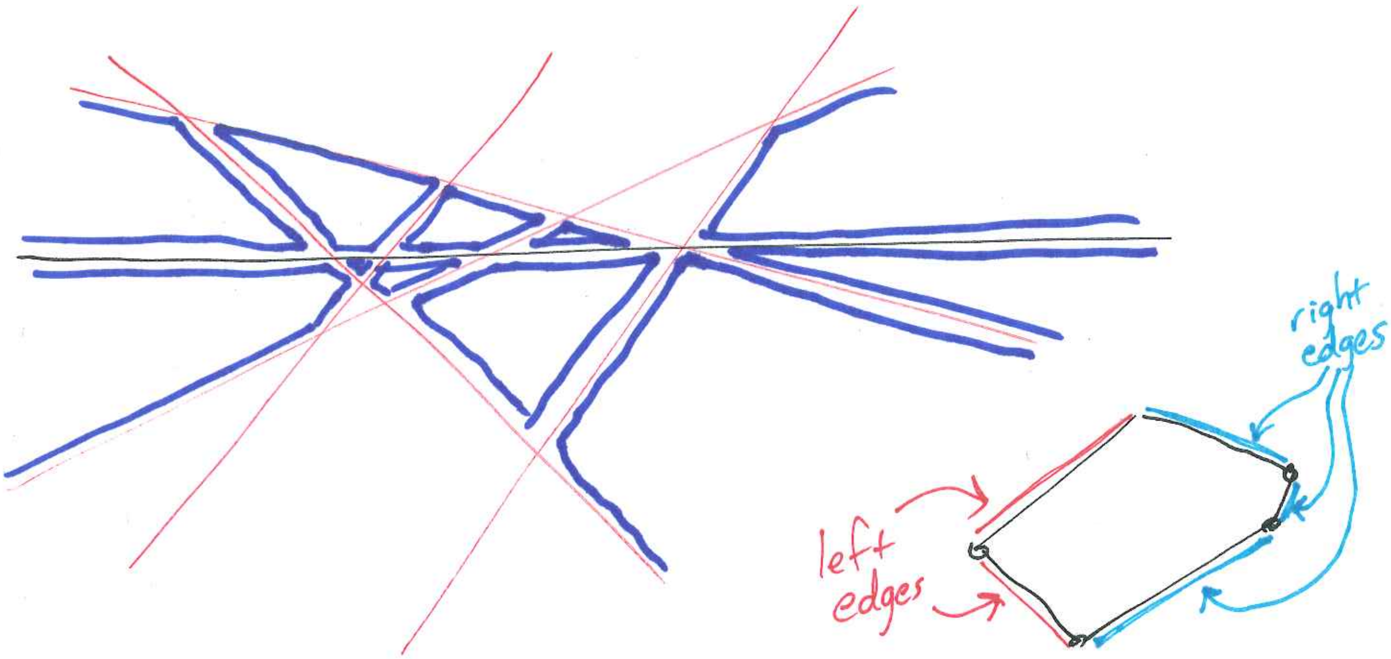
How long does this take?



We need to walk around the faces of every polygon in the arrangement that intersects l.

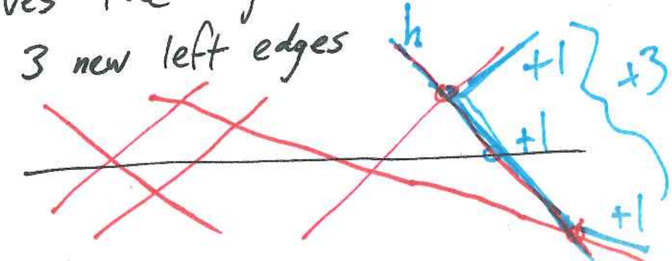
How many edges can be in this set of polygons?

Def The zone of a line ℓ in an arrangement $A(L)$ is the subcomplex formed by the polygons intersecting ℓ.

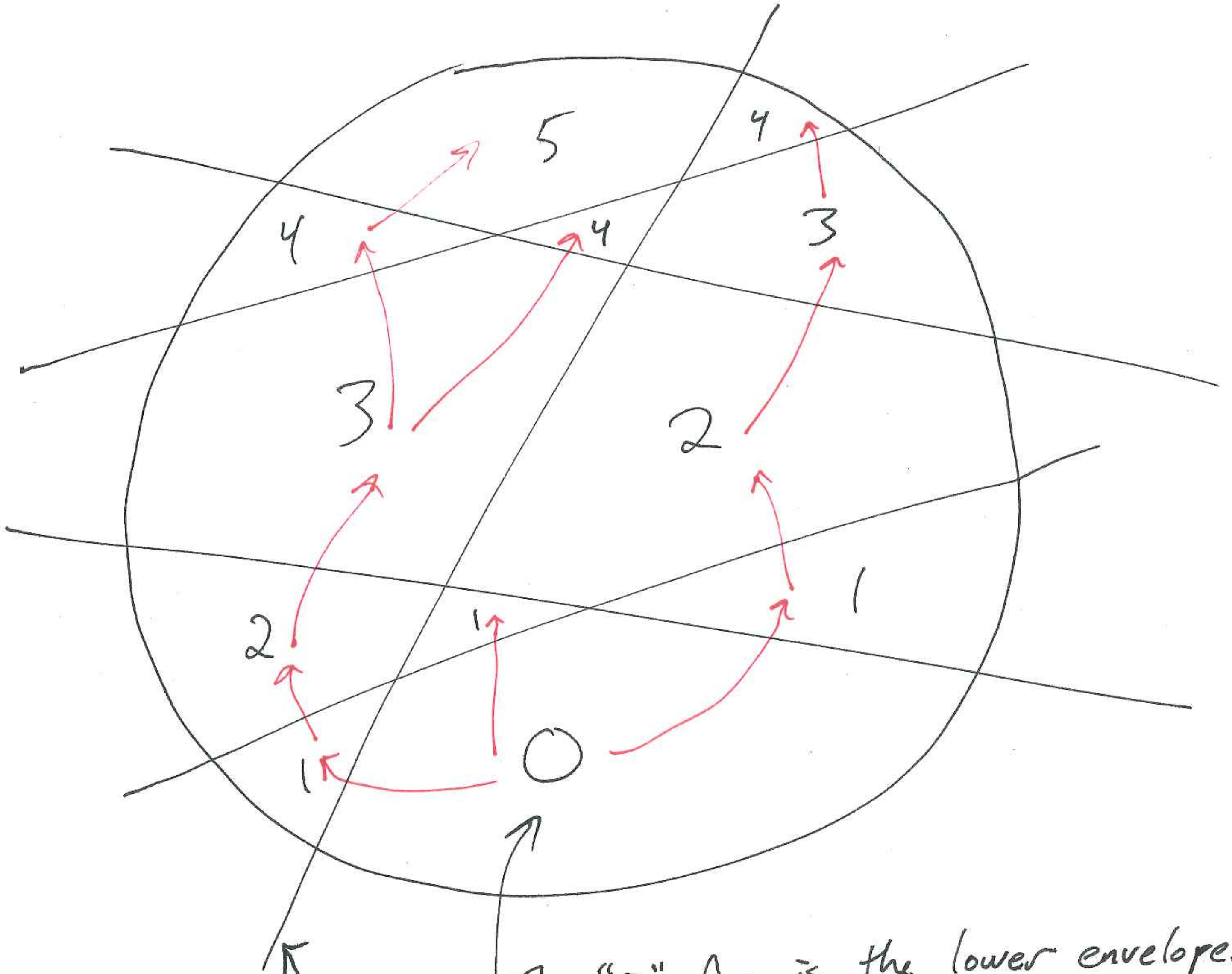


The Zone Thm Given n lines L, the zone of any $\ell \in L$ has at most $6n$ edges.

pf By induction on n.
WLOG assume ℓ is horizontal.
Suffices to show there are at most $3n$ "left" edges.
Ind. Hyp. For $n-1$ lines zone of ℓ has at most $3(n-1)$ left edges.
Remove the line h that gives the rightmost intersection with ℓ.
Adding h back in creates only 3 new left edges



How to add counts to the faces.



The "0" face is the lower envelope.
We can find one edge by taking the line with the largest slope.

Do a search through the dual of the arrangement. Each edge changes the count by one.

Takes time linear in the size of the arrangement. Running time: $O(n^2)$.

Summary

Halfspace Range Counting

Dualize

Build the Arrangement

Assign a count to every cell.

Use history DAG to find the right cell.

Preprocessing Time+Space: $O(n^2)$

Query Time: $O(\log n)$.